

# Sub-space approximations for MDO problems with disparate disciplinary variable dependence

Jonathan Ollar<sup>1,2</sup> · Vassili Toropov<sup>2</sup> · Royston Jones<sup>1</sup>

Received: 5 February 2016 / Revised: 24 April 2016 / Accepted: 23 May 2016 / Published online: 14 June 2016  
© The Author(s) 2016. This article is published with open access at Springerlink.com

**Abstract** An approach to solving multidisciplinary design optimisation problems using approximations built in sub-spaces of the design variable space is proposed. Each approximation is built in the sub-space significant to the corresponding discipline while the optimisation problem is solved in the full design variable space. Since the approximations are built in a space of reduced dimensionality, the computational budget associated with building them can be reduced without compromising their quality. The method requires the designer to make assumptions on which design variables are significant to each discipline. If such assumptions are deficient, the resulting approximations suffer from errors that are not possible to reduce by additional sampling. Therefore a recovery mechanism is proposed that updates the values of the insignificant variables at the end of each iteration to align with the current best point. The method is implemented within a trust region based optimisation framework and demonstrated on a multidisciplinary optimisation of a thin-walled beam section subject to stiffness and impact requirements.

**Keywords** Multidisciplinary design optimisation · Multi-model optimisation · Sub-space approximations · Mid-range approximations

## 1 Introduction

In multidisciplinary design optimisation (MDO) problems the individual disciplines are often evaluated using different computational models, making it not only MDO but also multi-model optimisation. At the same pace as computational performance is increasing, more complexity is built into the computational models, making them more computationally expensive. Thus when dealing with MDO problems one may face a very computationally expensive problem. For reviews on MDO architectures, see for instance (Cramer et al. 1993), Sobieszczanski-Sobieski and Haftka (1997), and most recently (Martins and Lambe 2013)

This work is focused on solving MDO problems using the multidisciplinary design feasible (MDF) architecture as formulated by Cramer et al. (1993) assisted by approximations. The reader is referred to Barthelemy and Haftka (1993), Wang and Shan (2007), Forrester and Keane (2009), and Viana et al. (2014) and Viana et al. (2010) for reviews on metamodel-based optimisation. The main challenge of using approximations is ensuring their quality. The number of training points required to obtain approximations of desired quality increases superlinearly with the number of design variables unless the responses are linear in the design variable space. For MDO problems the total number of design variables can become very large leading to the need for a large number of computationally expensive simulations.

Given that there are variables in the MDO problem that are not shared between all disciplines, i.e. have their own

---

Parts of this manuscript have been presented at the 15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, July, 2014, Atlanta, Georgia - Paper number: AIAA 2014-2437, and at the 56th AIAA/ASME/ASCE/AHS/ACS Structures, Structural Dynamics and Materials Conference, January, 2015, Kissimmee, Florida - Paper number: AIAA 2015-1585.

---

✉ Jonathan Ollar  
jonathan.ollar@gmail.com

<sup>1</sup> Altair Engineering Ltd, Imperial House, Holly walk, CV32 4JG, Leamington Spa, UK

<sup>2</sup> School of Engineering and Materials Science, Queen Mary University of London, Mile End Road, E1 4NS London, UK

relevant subsets of design variables, approximations can be built on only the sub-set of the design variables that is significant to each discipline whilst the optimisation problem is solved in the full design variable space. There are several examples of the use of this approach in the automotive industry, e.g. by Sobieszczanski-Sobieski et al. (2001), Kodiyalam et al. (2004), Ollar et al. (2014) and Ollar et al. (2015) and Ryberg et al. (2015). The benefit is that sampling and approximation building can be carried out in a space of reduced dimensionality. This can greatly reduced the required computational budget for obtaining an approximation of sufficient quality. However, deficient assumptions, made when identifying significant variables, lead to approximation errors that can not be reduced by additional sampling.

In this work, an adaptive discipline-related sampling and approximation building technique is introduced within the trust region framework known as the Mid-Range Approximation Method (MAM). In contrast with the previous work, it is part of a trust region strategy that accounts for erroneous assumptions by means of a recovery strategy that updates the values of the variables classed as insignificant but remain present in the full variable space during the optimisation stage of the trust region strategy.

## 2 Mid-range approximation method

The mid-range approximation method (MAM), also known as the multi-point approximation method, was originally reported by Toropov (1989, 1992) and Toropov et al. (1993). The MAM solves a typical constrained optimisation problem in the form

$$\begin{aligned} & \underset{\mathbf{x}}{\text{minimize}} \quad f_0(\mathbf{x}) \\ & \text{subject to} \quad f_j(\mathbf{x}) \leq 1, \quad j = 1, \dots, m \\ & \quad \quad \quad A_i \leq x_i \leq B_i, \quad i = 1, \dots, n \end{aligned} \quad (1)$$

where  $f_0(\mathbf{x})$  is the objective function,  $f_j(\mathbf{x})$  is the  $j$ -th constraint,  $\mathbf{x}$  is the vector of design variables and  $\mathbf{A}$  and  $\mathbf{B}$  are the upper and lower bounds respectively on the design variables. The optimisation problem (1) is replaced by a sequence of approximate sub-problems defined as

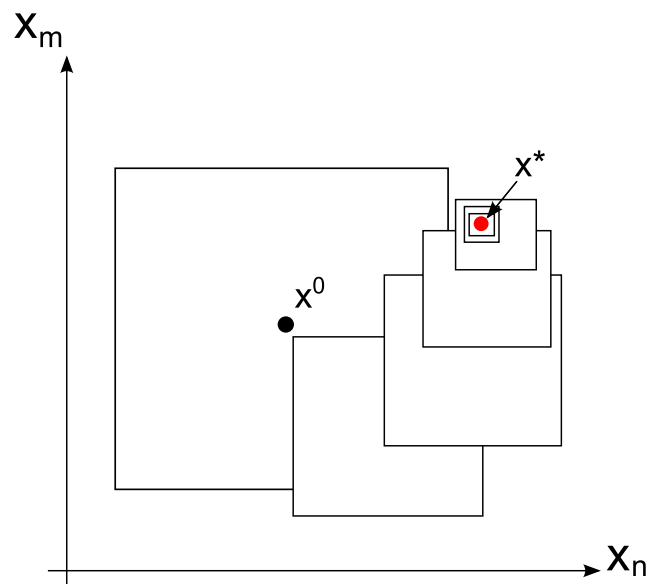
$$\begin{aligned} & \underset{\mathbf{x}}{\text{minimize}} \quad \tilde{f}_0^k(\mathbf{x}) \\ & \text{subject to} \quad \left. \begin{aligned} \tilde{f}_j^k(\mathbf{x}) &\leq 1, & j &= 1, \dots, m \\ A_i^k &\leq x_i \leq B_i^k \\ A_i^k &\geq A_i \\ B_i^k &\leq B_i \end{aligned} \right\} & i &= 1, \dots, n \end{aligned} \quad (2)$$

where  $k$  denotes the current iteration number,  $\tilde{f}_0^k(\mathbf{x})$  is an approximation of the objective function and  $\tilde{f}_j^k(\mathbf{x})$  is an approximation of the  $j$ -th constraint function, both considered valid only in the current trust region.  $A_i^k$  and  $B_i^k$  are the

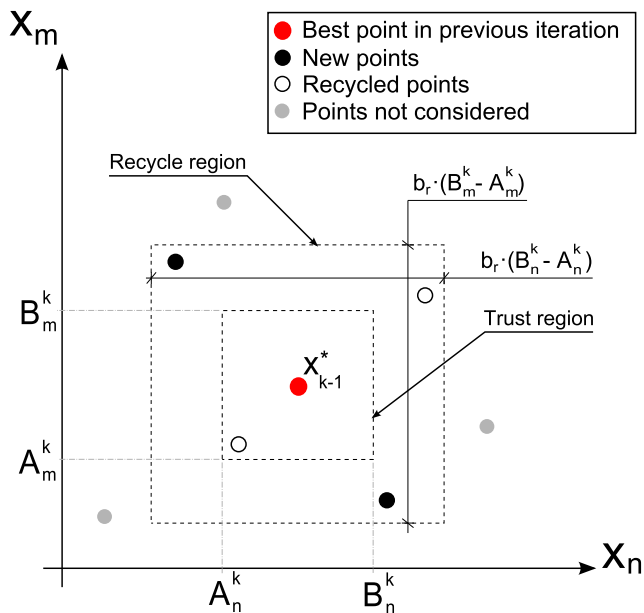
bounds of the current trust region where the sub-problem (2) is solved for the current iteration. The solution procedure for each sub-problem consists of sampling, creating approximations, solving the approximate optimisation problem and determining a new location and size of the trust region for the next iteration. The trust region will move and change size after each iteration until the termination criterion is reached. Figure 1 illustrates the history of trust regions through the sequence of sub problems in two dimensions. The trust region strategy has gone through several developments to account for the presence of numerical noise in the response function values (Van Keulen et al. 1996; Toropov et al. 1996) and occasional simulation failures (Toropov et al. 1999).

In the beginning of each iteration sampling, according to a small-scale design of experiments (DOE), is carried out within the trust region. Available sampling techniques include a technique based on extensible lattice sequences proposed by Hickernell et al. (2000) and a non-collapsible randomised technique by Korolev et al. (2015). The sampling process benefits from considering existing points, evaluated in previous iterations, however, points that are far away from the trust region may spoil the approximation and are not considered. The region in which points are considered is referred to as the *recycle region* and is defined as an enlargement of the trust region according to Fig. 2.

Approximations are created for each response using the evaluated sampling points. Available approximation



**Fig. 1** Typical history of the trust regions. In every iteration of the optimization process the new trust region is centered around the current solution and either kept the same size, reduced or enlarged



**Fig. 2** Points from previous iterations are re-used if they are located within the recycle region defined as an enlargement of the trust region by a factor  $b_r$

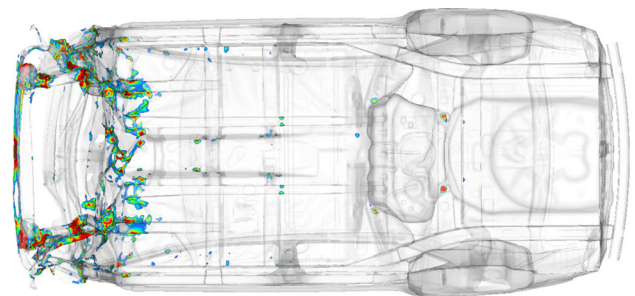
techniques include approximation assemblies by Polynkin and Toropov (2012), where intrinsically linear and rational functions are assembled into a single approximation using linear regression, the moving least squares method (Lancaster and Salkauskas 1981) and kriging based on a computationally efficient hyper-parameter optimisation method by Mortished et al. (2016).

The approximate sub-problem (2) is solved using the sequential quadratic programming (SQP) method developed by Madsen et al. (2002), based on the work of Powell (1985), starting from several points in order to increase the chance of finding a good solution for multi-modal problems. Candidate solutions of the approximated sub-problem (2) are evaluated in parallel with additional sampling points as introduced by Korolev et al. (2015) in order to fully utilise parallel high performance computing hardware. The best candidate point is compared to the previously obtained best point and, if superior, stored as the current best point.

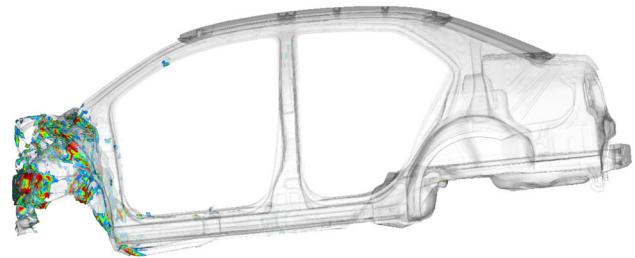
Finally, termination criteria are checked. The current best point may not be located at the bound of the trust region, the approximations must be of sufficient quality, and the trust region must be sufficiently small compared to the initial design region. If the criteria are met, the process is terminated. Otherwise the position of the trust region is moved so that its centre point coincides with the current best point. The trust region strategy changes the size of the trust region depending on the same factors as the termination criteria and the process is repeated.

### 3 Sub-space approximations

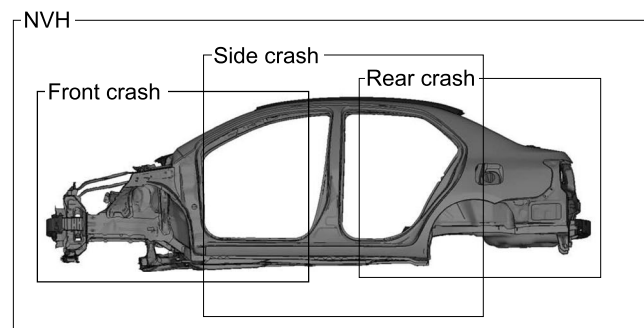
Suppose that the responses belonging to a discipline in the MDO problem only depend on a subset of the full set of design variables, i.e. a set of the variables has none or very little influence on the responses of the particular discipline. An example of this from the automotive industry can be seen in Fig. 3. The figure shows a simulation of a vehicle subjected to a front crash load case. Each element is coloured according to its level of internal energy. It can be concluded that, as can be expected, the energy absorption is concentrated in the front of the vehicle and is not much affected by the rest of the structure. A conceptual partitioning based on design variable dependence for four common automotive disciplines, *Front Crash*, *Side Crash*,



(a) Top view



(b) Side view



(c) Conceptual partitioning

**Fig. 3** Front crash simulation of an automotive model. Each element is coloured according to its level of internal energy. The model was developed by the National Crash Analysis Center (NCAC), The George Washington University, Washington, USA

Rear Crash and Noise Vibration and Harshness (NVH) are shown in Fig. 3c. With this partitioning discipline-related approximations, hereafter denoted sub-space approximations, can be built, depending only on the set of significant variables for each discipline.

### 3.1 Formulation of sub-space approximations

Here a mathematical formulation for introducing sub-space approximations in approximation assisted MDO is given. Unlike in previous work, all response functions are assumed to be defined in the full variable space of the optimisation problem in order to control insignificant variables. It will be shown that when sub-space approximations are used within a trust region framework, this becomes necessary to account for possible deficient assumptions on partitioning.

Consider solving the optimisation problem (1) using approximations. The optimisation problem becomes

$$\begin{aligned} & \underset{\mathbf{x}}{\text{minimize}} \quad \tilde{f}_0(\mathbf{x}) \\ & \text{subject to} \quad \tilde{f}_j(\mathbf{x}) \leq 1, \quad j = 1, \dots, m \\ & \quad \quad \quad A_i \leq x_i \leq B_i, \quad i = 1, \dots, n \end{aligned} \quad (3)$$

where  $\tilde{f}_0(\mathbf{x})$  is an approximation of the objective function and  $\tilde{f}_j(\mathbf{x})$  is an approximation of the  $j$ -th constraint function.

Given that the design variables in the optimisation problem can be categorised either as significant or insignificant for each related response. A projection can then be defined, for each response  $j$ , from the design variable space onto the space of the significant variables for that particular response. This is denoted as

$$\left. \begin{aligned} \xi_j &= P_j^\xi \mathbf{x} \\ P_j^\xi : \mathbb{R}^n &\mapsto \mathbb{R}^{s_j} \end{aligned} \right\}, \quad j = 0, \dots, m \quad (4)$$

where  $n$  is the number of design variables in the optimisation problem and  $s_j$  is the number of significant variables for the response  $j$ . A projection onto the space of the insignificant variables is defined in the same manner as

$$\left. \begin{aligned} \psi_j &= P_j^\psi \mathbf{x} \\ P_j^\psi : \mathbb{R}^n &\mapsto \mathbb{R}^{n-s_j} \end{aligned} \right\}, \quad j = 0, \dots, m. \quad (5)$$

From here on the projections are described according to the following convention

$$\mathbf{x} = \begin{bmatrix} \xi_j \\ \psi_j \end{bmatrix}, \quad j = 0, \dots, m, \quad (6)$$

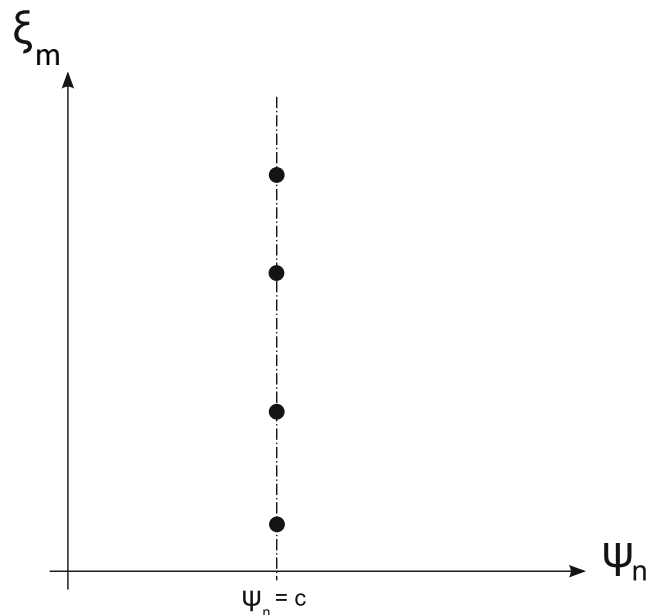
noting that the components of  $\xi_j$  and  $\psi_j$  are present in  $\mathbf{x}$  in arbitrary order. The responses in the optimisation problem can then be described as

$$f_j(\mathbf{x}) = f_j\left(\begin{bmatrix} \xi_j \\ \psi_j \end{bmatrix}\right), \quad j = 0, \dots, m \quad (7)$$

where the values of  $\psi_j$  can be chosen arbitrarily since they are deemed to be insignificant to the response. The approximations of the responses, therefore, may now be defined in the space of only the significant variables which allows a re-writing of the approximate optimisation problem according to

$$\begin{aligned} & \underset{\xi}{\text{minimize}} \quad \tilde{f}_0(\xi_0) \\ & \text{subject to} \quad \tilde{f}_j(\xi_j) \leq 1, \quad j = 1, \dots, m \\ & \quad \quad \quad \xi_j = P_j^\xi \mathbf{x}, \quad j = 0, \dots, m \\ & \quad \quad \quad A_i \leq x_i \leq B_i, \quad i = 1, \dots, n \end{aligned} \quad (8)$$

where each approximated response is defined only in the space of variables that are significant to the response. The optimisation problem, however, is defined in the full design variable space. This has the benefit that as each approximation is defined only in the space of the significant variables, the sampling of training points only needs to be carried out in that space while the values of the insignificant variables are kept constant as demonstrated by Fig. 4. Hence if the number of significant variables is small compared to the number of design variables, the density of the training points will increase leading to a better quality approximation as compared to what would have been achieved otherwise. Note that even though there is one projection per response, practicalities may require groups of responses to use the same projection, e.g. due to several responses being evaluated from the same simulation, or even responses belonging to different responses but connected through multi-physics coupling such as fluid structure interaction.



**Fig. 4** Sampling shown in two dimensions. Vertical axis corresponds to the significant variable  $\xi_m$  and horizontal axis to the insignificant variable  $\psi_n$ . Sampling is carried out in the space of the significant variable while the insignificant variable is kept at a constant value,  $c$

### 3.2 Sub-space approximations in trust regions

In this section an approach to building sub-space approximations within the MAM is proposed. A recovery mechanism for erroneous identification of significant variable for the sub-space partitioning is also suggested. Sub-space approximations can be introduced in the MAM framework by re-writing the sequence of optimisation problems (2) as:

$$\begin{aligned} & \underset{\mathbf{x}}{\text{minimize}} \quad \tilde{f}_0^k(\xi_0) \\ & \text{subject to} \quad \tilde{f}_j^k(\xi_j) \leq 1, \quad j = 1, \dots, m \\ & \quad \quad \quad \xi_j = P_j^\xi \mathbf{x}, \quad j = 0, \dots, m \\ & \quad \quad \quad \left. \begin{aligned} A_i^k &\leq x_i \leq B_i^k \\ A_i^k &\geq A_i \\ B_i^k &\leq B_i \end{aligned} \right\} \quad i = 1, \dots, n \end{aligned} \quad (9)$$

noting that the mid-range approximations created in each iteration are now functions of the significant variables only. The significant variables for each discipline are identified by the designer. Such judgement may be based on, for instance, engineering experience or design variable ranking studies. In the simplest form of sub-space approximations, used in the work by (Sobieszcanski-Sobieski et al. 2001), Kodiyalam et al. (2004), Ollar et al. (2014) and Ryberg et al. (2015), deficiencies in sub-space partitioning, i.e. by failing to identify significant variables, can result in approximation errors that cannot be resolved by additional sampling. This is due to changes in response values as a consequence of changes in the insignificant variables. Such changes can not be included in the approximations as they are not defined in the space of the insignificant variables.

Regardless of how carefully the partitioning of variables is made, there is always a risk that significant variables will be incorrectly identified as insignificant. Therefore a recovery mechanism for such errors is needed. This can be implemented in the trust region strategy by making sure that the vector of insignificant variables for the individual response is updated at the end of each iteration depending on the current best point as proposed by Ollar et al. (2015).

Let  $\mathbf{x}_{k-1}^*$  denote the solution vector to the previous iteration ( $k - 1$ ) for the optimisation problem (9). Recalling (4) and (5), the projection of the current best point onto the space of the significant variables can be written as

$$\xi_j^* = P_j^\xi \mathbf{x}^*, \quad j = 0, \dots, m \quad (10)$$

and onto the space of the insignificant variables as

$$\psi_j^* = P_j^\psi \mathbf{x}^*, \quad j = 0, \dots, m \quad (11)$$

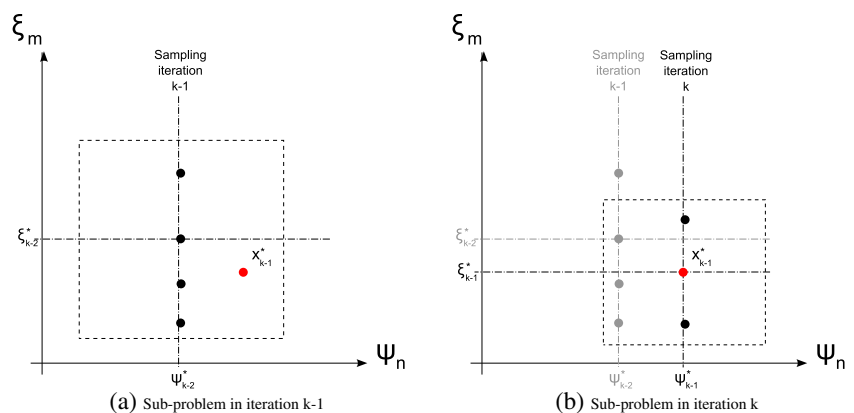
Any change in the response values as a consequence of changes in the insignificant variables  $\psi_j^*$  will not be accounted for by the approximations and will mean that even if the approximations are of excellent quality in the space in which they are defined, there will be a discrepancy between the result they deliver and the one returned by the verification of the current best point. In this work this discrepancy is recovered in the following iteration by setting the constant value of the insignificant variables during sampling according to the current best solution according to

$$\psi_k = \psi_{(k-1)}^* \quad (12)$$

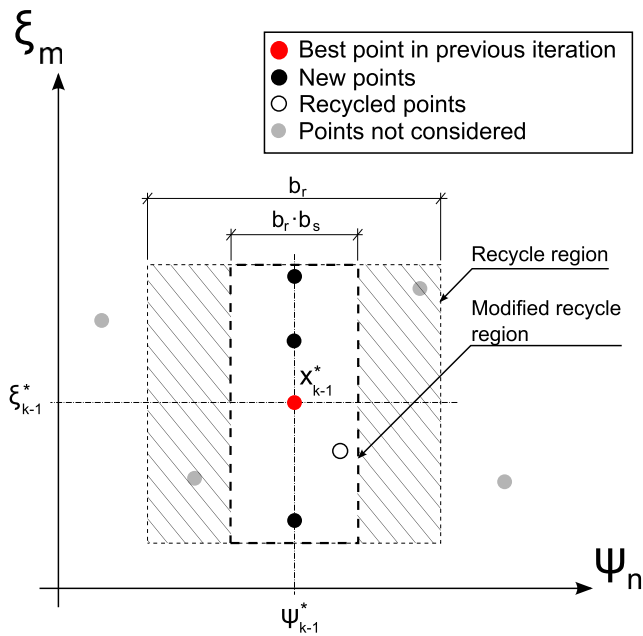
where  $\psi_k$  denotes the values of insignificant variables during sampling in the current iteration and  $\psi_{(k-1)}^*$  the current best solution in the previous iteration. Subscript  $j$  denoting the response number has been omitted for brevity. Figure 5 demonstrates how the value of  $\psi_n$  changes from the previous iteration,  $k - 1$ , to the current iteration,  $k$ , for the two dimensional case. As the approximations in the current iteration are built with the values of the insignificant variables updated according to the current best solution, any changes in response values due to changes in insignificant variables from the previous iteration will be taken into account by the approximations in the current iteration.

With the possibility of significant variables being identified as insignificant, there is a risk that existing points located within the recycle region but far away from the current value of the insignificant variables may spoil the resulting approximation. The recycle-region size is therefore multiplied by a reduction factor  $b_s$ , for the insignificant variables, as shown in Fig. 6.

**Fig. 5** The values of the insignificant variables in the solution of iteration  $k - 1$  differ from the same values in the sampling of iteration  $k-1$ . Potential changes in the function values as a consequence of this is taken into account in iteration  $k$  by updating the values of the insignificant variables for sampling according to the solution of iteration  $k - 1$







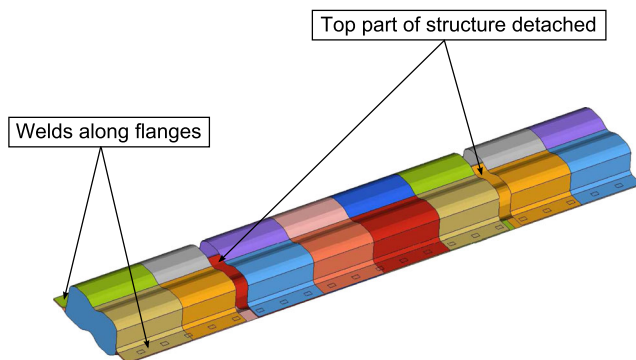
**Fig. 6** The recycle region (see Fig. 2) is reduced for insignificant variables by multiplication by a factor of  $b_s$

#### 4 Demonstration on a finite element model

This section describes a simple benchmark MDO study of a thin-walled beam structure subject to static and impact requirements. The impact requirements have disparate but overlapping variable dependence which makes it suitable for demonstration of the proposed sub-space approximation approach.

##### 4.1 Beam model

The beam is 1.2 m long and is made of two thin walled hat sections welded together along two flanges as shown in Fig. 7. It is modelled using shell elements with an average mesh size of 10 mm. The welds are modelled using single



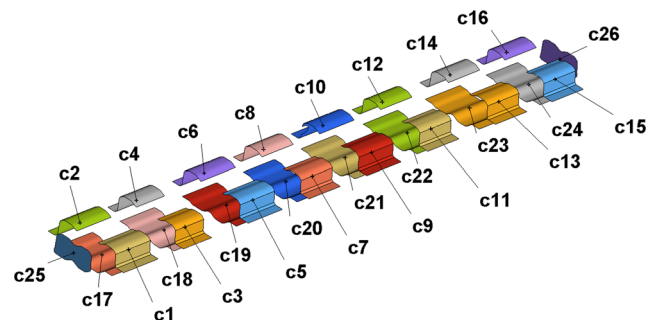
**Fig. 7** The thin-walled beam consisting of two hat sections spot-welded together along the flanges. Top part of structure has been detached at two locations

hexahedron elements connected to the shell elements using a fixed contact formulation. The structure is divided into 26 components, shown in Fig. 8, with individual thickness values which are to be determined. The starting thickness is 3 mm for each component and the allowable thickness range is 1–5 mm.

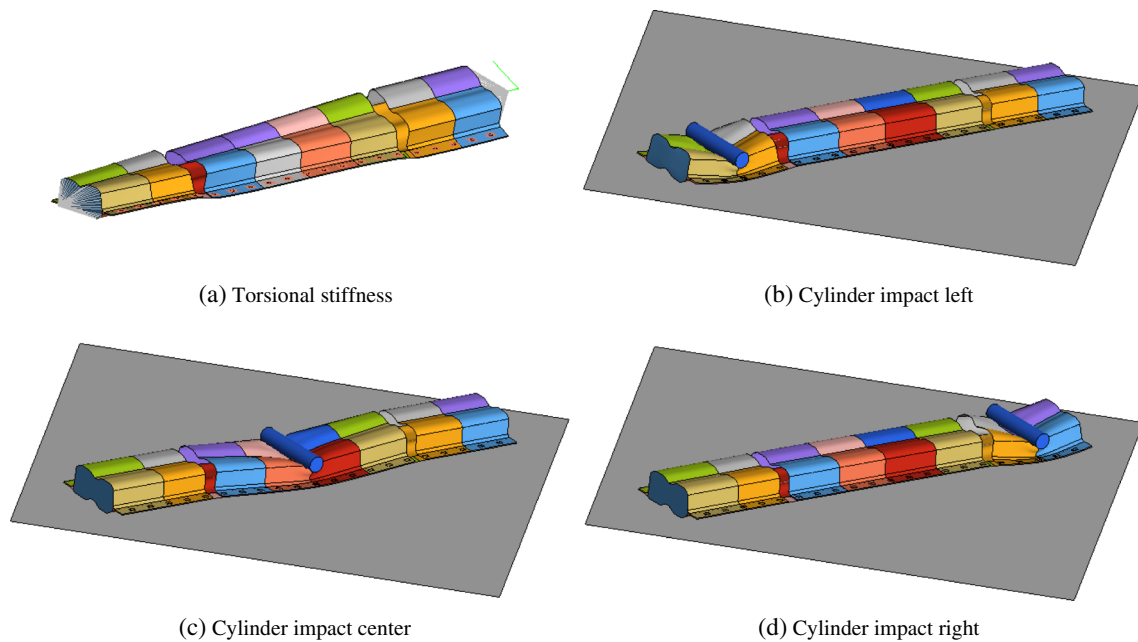
##### 4.2 Load cases

The beam is subjected to one static load case and three dynamic load cases as outlined below and shown in Fig. 9.

- Torsional stiffness** A free node on the lefthand side of the beam is constrained in all translational degrees of freedom as well as from rotation around the longitudinal axis of the beam. The node is connected to the edges of component c25 (see Fig. 8) using one dimensional elements (RBE3) that distributes forces from the free node to the structure. At the right hand side of the beam a force is applied to a lever arm, connected to the beam with the same type of connection, resulting in a moment around the longitudinal axis of the beam. The response is the resulting rotation at the point where the moment is applied. The mass of the structure is also used as a response and is extracted from this load case. The load case is analysed using a static implicit solution procedure in Altair OptiStruct (Altair Engineering, Inc., 2014a).
- Cylinder impact left (CL)** A heavy cylinder with an imposed velocity impacts the left section of the beam in vertical direction. The beam is supported in vertical direction by a rigid plane. At impact the beam is compressed by the heavy cylinder. The response is measured as the vertical deformation of the top part of the beam, caused by the cylinder impact. The load case, which is a dynamic explicit load case, is analysed using Altair RADIOSS (Altair Engineering, Inc., 2014b).
- Cylinder impact center (CM)** The load case definition is the same as for (b) except for the impact position of the cylinder which is now at the centre of the beam.



**Fig. 8** Exploded view of the 26 panels included in the MDO. Every panel has designable thickness range between 1–5 mm



**Fig. 9** Load cases included in the MDO

- (d) *Cylinder impact right (CR)* The load case definition is the same as for (b) and (c) except for the impact position of the cylinder which is now at the right section of the beam.

#### 4.3 Optimisation problem set up

The objective of this MDO problem is to minimise the mass of the beam subject to sufficient torsional stiffness and not exceeding maximum intrusion from the cylinder impact. The target values are summarised in Table 1 together with initial values, which are normalised to target, for each response. The mass and torsion response have available gradients which are used during the optimisation to build gradient-enhanced approximations. As the mass response is linear in the space of the design variables a simple linear regression using the least squares method (LSM) is used to approximate this response. The remaining responses are approximated using the moving least squares method (MLSM).

**Table 1** Summary of attributes for each response

Response	Initial	Target	Gradients	Approximations
Mass	1.00	<i>min</i>	Yes	LSM
Torsion	0.98	$\leq 1.00$	Yes	MLSM
CL	1.18	$\leq 1.00$	No	MLSM
CM	1.18	$\leq 1.00$	No	MLSM
CR	1.17	$\leq 1.00$	No	MLSM

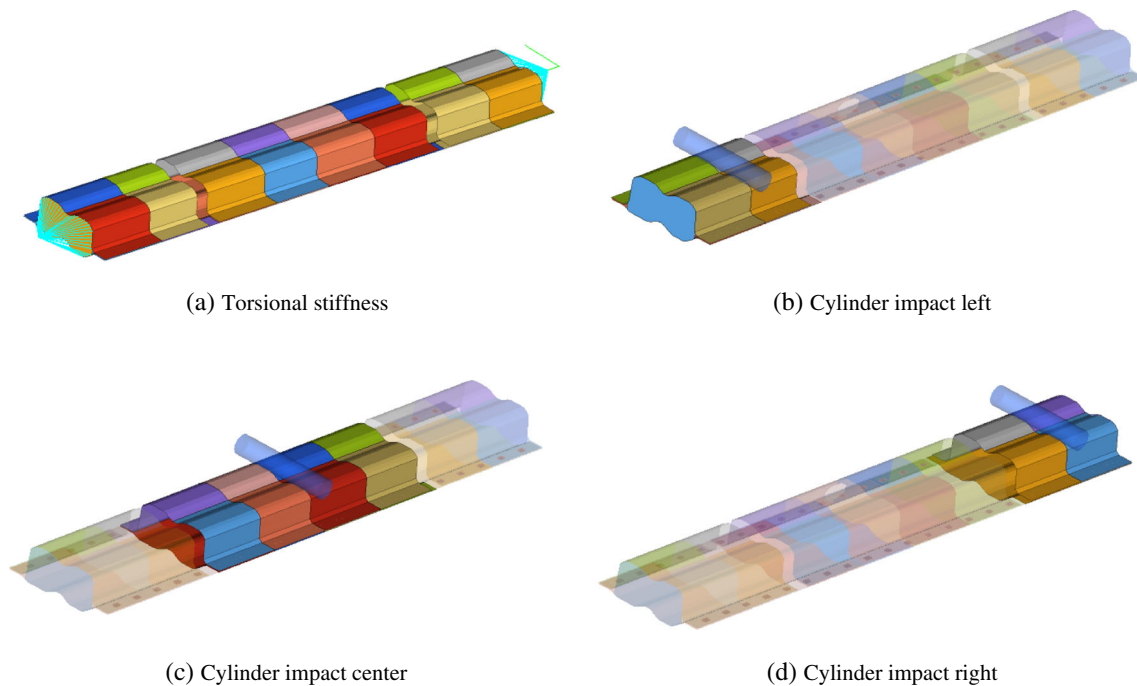
Four types of optimisations were carried out as outlined below. All optimisations were carried out using the MAM with 10 candidate points in each iteration obtained as the best out of 20 SQP optimisations starting from different points, sampled within the trust-region. The number of significant variables identified for each discipline and the corresponding number of points per iteration is presented in Table 2 for each optimisation. The convergence criteria was set such that the approximation quality must be below 5 % and the trust region size must be smaller than 10 % of the design region. The DOE method used for sampling of training points and start points for the SQP is based on a random number generator and will hence produce a different set of points depending on the initially chosen seed. In order to account for this uncertainty 50 optimisations with varying seed were carried out for each of the optimisation types described below.

**Table 2** No. points per iteration for each optimisation

Opt.	1		2		3		4	
	ns	np	ns	np	ns	np	ns	np
Torsion	26	39	26	39	26	39	26	39
CL	26	39	7	39	7	10	7	10
CM	26	39	12	39	12	18	11	18
CR	26	39	7	39	7	10	7	10

ns - number of significant variables

np - number of points per iteration



**Fig. 10** Sub-space partitioning for the beam structure. Insignificant variables are showed as transparent

#### 4.3.1 Optimisation 1 - Full space approximations

The first optimisation is set up as a point of comparison using the MAM without sub-space approximations. The number of points evaluated per iteration is chosen as  $np = 1.5 \times n = 39$  per load case, leading to a total of 156 points per iteration.

#### 4.3.2 Optimisation 2 - Same budget

In the second optimisation sub-space approximations with the proposed recovery mechanism is used within the MAM. Partitioning of the beam model is shown in Fig. 10, where insignificant variables are shown transparent. The optimisation is carried out with the same computational budget per iteration as the first iteration in an attempt to increase the quality of the approximations and hence a greater chance of the trust region moving in the right direction early on in the optimisation, leading to fewer iterations required for convergence.

#### 4.3.3 Optimisation 3 - Decreased budget

In the second optimisation sub-space approximations with the proposed recovery mechanism is used within the MAM. The partitioning of variables is the same as in the second optimisation. Here the number of points are determined individually for each discipline as  $ns = 1.5 \times ns$ , where  $ns$  is the number of significant variables for the discipline. The

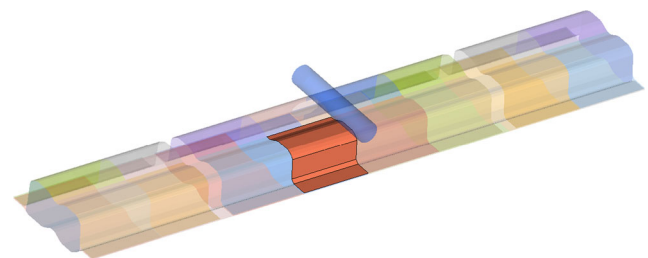
aim of the optimisation is to keep the number of required evaluations per iteration to a minimum.

#### 4.3.4 Optimisation 4 - Erroneous partitioning

In the final optimisation sub-space approximations with the proposed recovery mechanism is used within the MAM. The partitioning of variables is the same as in the second and third optimisation with one exception. Variable c7, shown in Fig. 11, which is significant to discipline CM, is deliberately identified as insignificant in order to test the proposed recovery mechanism outlined in Section 3.2.

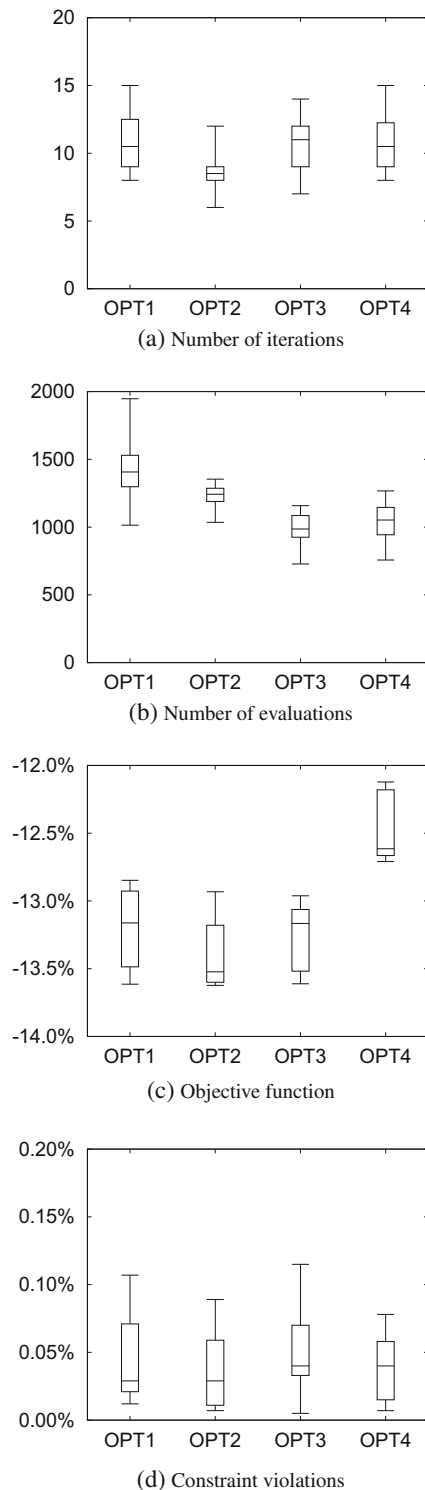
## 4.4 Results

The results of the study is presented in Fig. 12. For each optimisation the median, upper and lower quartiles and the minimum and maximum value is shown for the number of



**Fig. 11** Significant variable erroneously identified as insignificant





**Fig. 12** Statistical results from 50 runs with varying seed of each optimisation type, 1-4

iterations, number of evaluations, the objective function and the maximum constraint violation.

The first optimisation which was used as a comparison for the other three optimisations finished on average in 10.5

iterations, having spent 1407 evaluations, and with an average reduction in objective function of 13.2 %. The second optimisation, which was using sub-space approximations with the same number of evaluations per iteration as the first optimisation, finished on average in 8.5 iterations, 2 iterations less than the first, having spent 1242 evaluations, 165 less than the first optimisation, and with an average reduction in the objective function of 13.5 %, 0.3 % more than the first optimisation. The third optimisation, which was carried out using sub-space approximations with individual allocation of the number of points per iteration for each discipline, finished on average in 11 iterations, slightly more than the first optimisation but having spent 986 evaluations, 421 less than the first optimisation and 256 less than the second. The average reduction in objective function was 13.2 %, just like in the first optimisation. The fourth and final optimisation, where a significant variable was deliberately identified as insignificant, finished on average after 10.5 iterations, having spent 1052 evaluations, with an objective reduction of 12.6 %, 1 % less than the first optimisation.

It can be concluded from the results that, by using sub-space approximations for the presented example, it is possible to reduce the number of required iterations and/or the number of evaluations for carrying out the optimisation without compromising the results. By maintaining the number of points that are needed for full space optimisation when using sub-space approximations, as in optimisation 2, the number of iterations can be reduced. If instead, the number of evaluations per iteration is individually allocated, as in optimisation 3, for each discipline, a reduction in the number of evaluations can be reduced. It can also be included from the fourth optimisation that the proposed recovery mechanism makes sure that erroneous identification of significant variables does not lead to constraint violations, but instead to a slight penalty in the objective function.

## 5 Conclusions

An approach to carrying out multidisciplinary design optimisation using approximations built in individual sub-spaces for each discipline while carrying out the optimisation in the full variable space has been proposed. The sub-spaces in which the approximations are built are defined by the sets of significant variables for the individual disciplines. The main benefit of the technique is the dimensionality reduction of the approximations and sampling. This enables reducing the computational budget required to obtain approximations of sufficient quality.

The method relies on the designer to make assumptions on which variables are significant for each response. If such assumptions are deficient, approximation errors can occur

that cannot be reduced by additional sampling. Therefore a technique was proposed that can recover from such errors within a trust region based optimisation framework. By updating the values of the variables identified as insignificant, but remain present in full space, according to the best current solution in each iteration, the technique can recover from such errors.

The approach was demonstrated on a finite element example of a thin-walled beam. A reduction in the computational budget was shown for optimisations carried out using the sub-space approximation approach compared to conventional optimisations. In addition, a test where a significant variable was deliberately identified as insignificant was carried out and showed the validity of the developed recovery mechanism.

**Acknowledgments** The research leading to these results have been funded by the European Union Seventh Framework Programme FP7-PEOPLE-2012-ITN under grant agreement 316394, Aerospace Multidisciplinarity Enabling DEsign Optimization (AMEDEO) Marie Curie Initial Training Network.

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

## References

- Altair Engineering Inc (2014a) Optistruct 13.0 user's guide. <http://www.altairhyperworks.com>, accessed: 2014-12-01
- Altair Engineering Inc (2014b) Radioss 13.0 user's guide. <http://www.altairhyperworks.com>, accessed: 2014-12-01
- Barthelemy JFM, Haftka RT (1993) Approximation concepts for optimum structural design — a review. *Struct Optim* 5(3):129–144
- Cramer EJ, Dennis Jr JE, Frank PD, Lewis RM, Shubin GR (1993) Problem formulation for multidisciplinary optimization. *SIAM J Optim* 4:754–776
- Forrester AI, Keane AJ (2009) Recent advances in surrogate-based optimization. *Progress Aerosp Sci* 45(1–3):50–79
- Hickernell FJ, Hong HS, Lécuyer P, Lemieux C (2000) Extensible lattice sequences for quasi-Monte Carlo quadrature. *SIAM J Sci Comput* 22(3):1117–1138
- Kodiyalam S, Yang R, Gu L, Tho CH (2004) Multidisciplinary design optimization of a vehicle system in a scalable, high performance computing environment. *Struct Multidiscip Optim* 26(3–4):256–263
- Korolev Y, Toropov V, Shahpar S (2015) Large-scale CFD optimization based on the FFD parametrization using the multipoint approximation method in an HPC environment. Paper AIAA 2015-3234, 16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA Aviation, Dallas, TX, pp 22–26
- Lancaster P, Salkauskas K (1981) Surfaces generated by moving least squares methods. *Math Comput* 37(155):141–158
- Madsen K, Nielsen HB, Sondergaard J (2002) Robust subroutines for non-linear optimization. University of Denmark IMM-REP-2002-02
- Martins JRRA, Lambe AB (2013) Multidisciplinary design optimization: a survey of architectures. *AIAA J* 51(9):2049–2075
- Mortished C, Ollar J, Jones R, Benzie P, Toropov V, Sienz J (2016) Aircraft wing optimisation based on computationally efficient gradient-enhanced kriging. 57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Kissimmee, FL, USA, January 4–8, 2015
- Ollar J, Toropov V, Jones R (2014) Mid-range approximations in sub-spaces for MDO problems with disparate discipline attributes. 15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, GA, USA, pp 16–20
- Ollar J, Toropov V, Jones R (2015) Adaptive sub-space approximations in trust-regions for large scale MDO problems. 56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, FL, USA, pp 5–9
- Polynkin A, Toropov V (2012) Mid-range metamodel assembly building based on linear regression for large scale optimization problems. *Struct Multidiscip Optim* 45(4):515–527
- Powell M (1985) On the quadratic programming algorithm of Goldfarb and Idnani. In: Cottle R (ed) *Mathematical programming essays in honor of george b. Dantzig Part II, Mathematical Programming Studies*, vol 25. Springer Berlin Heidelberg, pp 46–61
- Ryberg AB, Bckryd R, Nilsson L (2015) A metamodel-based multidisciplinary design optimization process for automotive structures. *Eng Comput* 31(4):711–728
- Sobieszcanski-Sobieski J, Haftka R (1997) Multidisciplinary aerospace design optimization: survey of recent developments. *Struct Optim* 14(1):1–23
- Sobieszcanski-Sobieski J, Kodiyalam S, Yang R (2001) Optimization of car body under constraints of noise, vibration, and harshness (NVH), and crash. *Struct Multidiscip Optim* 22(4):295–306
- Toropov V (1989) Simulation approach to structural optimization. *Struct Optim* 1(1):37–46
- Toropov V (1992) Multipoint approximation method in optimization problems with expensive function values. In: *Computational systems analysis 1992: Proceedings of the 4th International Symposium on Systems Analysis and Simulation*. v. 4. Elsevier
- Toropov V, Filatov A, Polynkin A (1993) Multiparameter structural optimization using FEM and multipoint explicit approximations. *Struct Optim* 6(1):7–14
- Toropov V, van Keulen F, Markine V, De Boer H (1996) Refinements in the multi-point approximation method to reduce the effects of noisy structural responses. 6th Symposium on Multidisciplinary Analysis and Optimization, Bellevue, WA, pp 4–6
- Toropov V, Markine V, Holden C (1999) Use of mid-range approximations for optimization problems with functions of domain-dependent calculability. 3rd ISSMO/UBCAD/UB/AIAA World Congress of Structural and Multidisciplinary Optimization, NY, USA, pp 17–21
- Van Keulen F, Toropov V, Markine V (1996) Recent refinements in the multi-point approximation method in conjunction with adaptive mesh refinement. *ASME Design Engineering Technical Conferences and Computers in Engineering Conference*, Irvine, CA, pp 18–22
- Viana FA, Gogu C, Haftka RT (2010) Making the most out of surrogate models: tricks of the trade. In: *ASME 2010 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers, pp 587–598
- Viana FAC, Simpson TW, Balabanov V, Toropov V (2014) Metamodeling in multidisciplinary design optimization: How far have we really come? *Progress Aerosp Sci* 52(4):670–690
- Wang GG, Shan S (2007) Review of metamodeling techniques in support of engineering design optimization. *J Mech Des* 129(4):370–380